

Extracting and Predicting Taxi Hotspots in Spatiotemporal Dimensions Using Conditional Generative Adversarial Neural Networks

Hao Yu ^{ID}, Zhenning Li ^{ID}, Guohui Zhang ^{ID}, Member, IEEE, Pan Liu, and Jun Wang, Member, IEEE

Abstract—It is of practical importance to extract and predict taxi hotspots in urban traffic networks. However, the extraction of taxi hotspot is generally influenced by multiple sources of dependence, which has not been well recognized in the existing literature. This study aims to investigate how the integration of clustering models and deep learning approaches can learn and extract the network-wide taxi hotspots in both temporal and spatial dimensions. A density based spatiotemporal clustering algorithm with noise (DBSTCAN) was established to extract the historical taxi hotspots, which changed with time. A conditional generative adversarial network with long short-term memory structure (LSTM-CGAN) model was proposed for taxi hotspot prediction, which is capable of capturing the spatial and temporal variations of taxi hotspots simultaneously. Specifically, the DBSTCAN was applied to process the large-scaled geo-coded taxi pickup data into time-varying historical hotspot information. The proposed LSTM-CGAN model was then trained by the network-wide hotspot data. As illustrated in the numerical tests, it was found that the proposed LSTM-CGAN model provided comparable results with different model layouts and model with 4 LSTM layers in both generator and discriminator performed best. Moreover, four typical prediction approaches were selected to compare with the proposed LSTM-CGAN model. The comparative analyses showed that the proposed LSTM-CGAN model reduced the false positive rate from 5.20% to 0.56% and the false negative rate from 63.50% to 39.93%, and improved the section consistency from 41.78% to 73.43% and the area under the curve from 0.670 to 0.969. The comparison results indicated that the proposed LSTM-CGAN model outperformed all these benchmark methods and demonstrated great potential to enable many shared mobility applications.

Index Terms—Taxi demand, conditional generative adversarial network, clustering, spatiotemporal correlation, taxi trajectory.

Manuscript received May 24, 2019; revised November 30, 2019 and February 10, 2020; accepted February 17, 2020. Date of publication March 4, 2020; date of current version April 16, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61803083, in part by the China Postdoctoral Science Foundation under Grant 2018M630497, and in part by the National Natural Science Foundation of China under Grant 61672236. The review of this article was coordinated by Prof. H. Chaoui. (*Corresponding author: Guohui Zhang.*)

Hao Yu, Zhenning Li, and Guohui Zhang are with the Department of Civil Environmental Engineering, University of Hawaii Manoa, Honolulu, HI 96822 USA (e-mail: seudarwin@gmail.com; li2016@hawaii.edu; guohui@hawaii.edu).

Pan Liu is with the School of Transportation, Southeast University, Nanjing 210096, China (e-mail: liupan@seu.edu.cn).

Jun Wang is with the School of Computer Science and Technology, East China Normal University, Shanghai 200241, China (e-mail: jwang@cs.ecnu.edu.cn).

Digital Object Identifier 10.1109/TVT.2020.2978450

I. INTRODUCTION

WITH the increasing population and scale of modern cities, it has been a paradox in urban traffic management, which is the unsatisfied mobility demands and the limited transportation resources, especially for taxi services. By the end of 2016, individual taxi traveled 204.02 km per day in Beijing, while the average empty cruising ratio was 40.2%. It implies that taxis run with low efficient in nearly two fifths time, which results in resource consumption. Meanwhile, more and more citizens are annoyed by the difficulty of taking a taxi [1]. The inefficient taxi distribution aggravates the paradox extensively. It is of practical importance to extract and predict the taxi hotspots in urban traffic networks.

With the increasing popularization of in-vehicle global positioning system (GPS) devices, taxi status, such as standing for a passenger, empty cruising, occupied, and off shift, is enhanced with spatiotemporal information, and high quality geocoded taxi information is accessible for analyzing and extracting taxi hotspot in both temporal and spatial dimensions [2]. Recently, identification of taxi hotspot has been a relatively new and exciting field of research [2]–[12]. Researchers have adopted either general clustering algorithms, such as K-means [3]–[6], clustering large application (CLARA) algorithm [6], and DBSCAN algorithm [5], [12]–[14], or taxi-based clustering algorithms, such as the density based hierarchical clustering (DBH-CLUS) method [9], to locate the high potential sections of taxi pickups. Those clustering-based approaches mainly focus on exploring the spatial correlation among taxi demand. In order to provide accurate demand information to taxi drivers in temporal dimension, several studies have been conducted on developing a time-varying recommending system [1]–[4], [15], [16]. However, the extracted taxi hotspots were usually fixed locations or changed periodically. Existing studies thus focused on optimal strategies combining the pre-determined taxi hotspots and real-time passenger/drivers requests.

Referred as a homogeneous mobility demand characteristic, identification of taxi hotspot is generally influenced by multiple sources of dependencies, including spatiotemporal dependency and external dependencies [17], [18]. Spatiotemporal dependency refers to the fact that the existing of taxi hotspots is affected not only by factors within current time and space, but also by the surrounding conditions in both past and present [19]. For example, recent studies usually applied the Euclidean or GPS

distance as a countermeasure of spatial correlation under the base assumption that characteristics of nearby spatial units make greater impacts than the distant ones do [20]. However, taxi demands are spatiotemporally interrelated through road network, which can hardly be captured by the Euclidean distance directly. A network-based distance is required for better understanding the spatiotemporal correlation among taxi demands and extracting the taxi hotspots. The external dependencies, such as weather conditions, and traffic regulations, may affect the taxi hotspots both spatially and temporally. Though efforts have been devoted to improving the efficiency of taxi operation, the homogeneous taxi mobility characteristic analysis in urban area is still an open issue and a great challenging. Limited methods have been applied to deal with these dependencies simultaneously. This study focused on both extracting and predicting taxi hotspots in spatiotemporal dimensions, which helps vacant taxis find the mobility demands.

Deep learning approaches have grown rapidly during the past few years in the research field of computer vision, pattern recognition, and artificial intelligence [21]. Recently, some studies have also applied the deep learning approaches into the field of traffic prediction, such as short-term traffic flow [22], [23], real-time traffic speed [24], network-wide traffic state [25], short-term crash risk [26], and passenger-demand for real-time ride service [11], [17], [27], [28]. Compared with traditional statistical models and other learning architectures, deep learning is capable of capturing complex non-linear relationships using hierarchical feature representation, which has exhibited its superiority in short-term traffic prediction [29], [30]. Therefore, the appearance of deep learning approaches has the potential to advance the understanding of taxi-based mobility demands and to enable the network-wide short-term taxi hotspot prediction.

The primary objective of this study is to develop a network-wide taxi hotspot extraction and prediction framework via an integration model of clustering and deep learning approaches. The contributions of this paper can be summarized as follows:

- A spatiotemporal equivalent distance is applied in the a density-based clustering approach to cluster geocoded taxi pickup objects into a list of spatiotemporal taxi hotspot indicators, so that spatial distribution and possible temporal correlation of homogeneous mobility characteristics is naturally captured.
- Inspired by Liang *et al.* [25], an adversarial architecture, namely the conditional generative adversarial network with long short-term memory structure (LSTM-CGAN), is modelled to capture the spatiotemporal distribution of taxi hotspots using the clustering results and corresponding information. And
- Validated by the taxi-based multi-source datasets collected in urban area in Beijing, China, the proposed hotspot extraction and prediction framework outperformed the selected benchmark methods, including traditional time-series methods and several state-of-the-art machine learning approaches.

The rest of this paper was organized as follows. Section II introduces the density-based spatiotemporal clustering algorithm with noise (DBSTCAN) and the LSTM-CGAN model. The

TABLE I
DESCRIPTION OF NOTATIONS

Type	Notation	Definition
Constant	S	set of spatial units
	R	set of temporal units
	P	set of taxi pickup objects
DBSTCAN	ε	neighborhood radius
	m_p	minimum required data objects number
	p_i	attribute vector for i th taxi pickup object
	l_{oi}	longitude for i th taxi pickup object
	l_{ai}	latitude for i th taxi pickup object
	t_i	time stamp for i th taxi pickup object
	$l_{(p_i, p_j)}$	targeted distance between p_i and p_j
	d_{k, p_i}	distance between node k and p_i
	$v_{(s, r)}$	average speed for spatiotemporal unit (s, r)
	L	shortest path matrix for taxi pickup objects
	$N_\varepsilon(p_i)$	set of objects within ε from p_i
LSTM-CGAN	G	generative structure in CGAN
	D	discriminator structure in CGAN
	x	true data
	y_x	conditional information corresponding to x
	z	random generate sample
	$f_p(s, r)$	hotspot indicator for spatiotemporal unit (s, r)
	P_z	prior noise distribution
	P_{data}	underlying distribution for true data
	l_G^P	loss function for generator G in pre-training
	l_D^P	loss function for discriminator D in pre-training
	l_G	loss function for generator G in training
	l_D	loss function for discriminator D in training

integration framework is described in Section III, including raw data processing, the clustering-based extracting part, and the deep learning based predicting part. In addition, the training procedure for the proposed LSTM-CGAN model is presented. Numerical case in the large-scale urban network is conducted to evaluate the performance of the integration framework in Section IV. In the end, we concluded the paper with discussions on future research in Section V.

II. METHODOLOGY

In this section, we introduced the DBSTCAN algorithm applied to the geocoded taxi pickup objects, and the LSTM-CGAN model. For convenience, key notations referred in the rest content were summarized in Table I.

A. DBSTCAN Algorithm

The density-based spatial clustering algorithm with noise (DBSCAN) was one of the most popular clustering algorithms in the past decades [13]. Compared to the other well-known clustering algorithms, such as the K-means clustering, DBSCAN applies the concept of reachability, i.e., how many neighbors have a point within a distance. DBSCAN does not rely on a pre-specified parameter K representing the number of clusters hidden in the dataset, and discovers clusters with arbitrary shapes using minimal domain knowledge required to determine the input parameters, which motivates the application in this study. A spatiotemporal enhanced DBSCAN, i.e., the DBSTCAN, was applied to capture the spatiotemporal correlation and enroute-paths in the taxi pickup dataset. Instead of Euclidean distance, a spatiotemporal distance between two taxi pickup objects was defined to quantify the relationship between two taxi pickup events. By simultaneously taking shortest path distance in road

network and the difference in time stamps into considering, the DBSTCAN algorithm thus grouped the taxi pickup objects into multiple spatiotemporal hotspots.

Let $P = \{p_1, \dots, p_m\}$ be the dataset containing m taxi pickup objects. Each object is described by an attribute vector, $p_i = (l_{oi}, l_{ai}, t_i, d_{n_1, p_i}, d_{n_2, p_i}, (s, r))$, where n_1 and n_2 are the two intersections at both ends of the link where the objects occurs; (s, r) is the index for the associated spatiotemporal unit. Let L be (conceptually only) a $m \times m$ matrix consisting of the target distances $l(p_i, p_j)$ among taxi pickup objects in a metric space. In coherence with the DBSCAN algorithm, we defined parameters as follows.

Definition 1 (Target Distance): The spatiotemporal distance between two pickups is defined as the target distance. It is made up of two quadrature components, which are the spatial distance and the temporal equivalent distance. The spatial distance refers to the shortest path distance connecting the two pickup locations in the road network, while the temporal equivalent distance is obtained by multiplying the time interval and average speed during the specific time slot.

Definition 2 (Core, Border and Noise Object): Object p_i is a core object, if its ε -neighborhood contains more than m_p objects, that is, if $|N_\varepsilon(p_i)| \geq m_p$, where $|\cdot|$ is the cardinality. Object p_j is determined as a border object if: (1) $|N_\varepsilon(p_j)| < m_p$; (2) $|N_\varepsilon(p_i)| \geq m_p$; and (3) $p_j \in N_\varepsilon(p_i)$. And object p_k is determined as a noise object if it is not a core or a border object.

Definition 3 (Spatiotemporal Unit): Urban traffic network is divided into uniform road sections, namely the spatial unit, while time horizon for the whole dataset is divided into multiple time slots. Thus, s th spatial unit in r th time slot is termed as the spatiotemporal unit (s, r) .

In general, the length of spatial unit shall be selected carefully. A large spatial unit may not capture the spatial homogeneity well, however, a small spatial unit significantly increases the computation. In the present study, each link is divided into several 250-meter sections and each section is termed as a spatial unit. As a result, the road network is divided into 11153 spatial units. A duration of 10–15 min for aggregating traffic demand is recommended in many previous studies [31]–[33]. In the presented study, 10 min span is selected to be a time slot, which maximizes the sample size.

Definition 4 (Taxi Hotspot): A taxi hotspot is a spatiotemporal unit who contains at least one core object.

A taxi hotspot indicator $f_p(s, r)$ is proposed accordingly. More specifically, taxi hotspot indicator $f_p(s, r)$ equals 1 if spatiotemporal unit (s, r) is a taxi hotspot. Otherwise, $f_p(s, r)$ equals 0. The new definitions imply that the DBSTCAN is formed based on a symmetric notion of reachability. Thus, the clustering algorithm is almost the same as DBSCAN, except for the calculation of target distance.

B. LSTM-CGAN Model

Introduced as a novel approach in generative model training, generative adversarial network (GAN) has been one of the pioneering DL technologies [25], [34]–[36]. The proposed

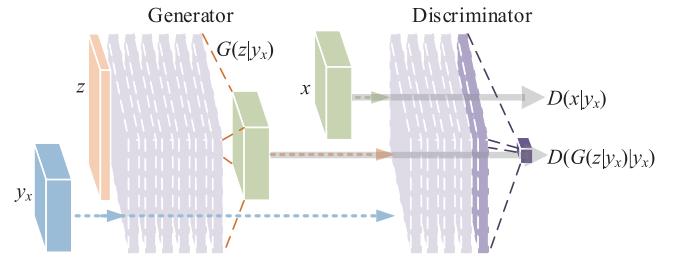


Fig. 1. Conditional architecture applied in the proposed LSTM-CGAN model.

LSTM-CGAN model enables the GAN model to cope with conditional information and LSTM technique.

The basic adversarial architecture in the LSTM-CGAN model consists of the generative structure G and the discriminative structure D . More specifically, the generator G captures the data distribution, and the discriminator D estimates the probability that a sample come from the true data rather than generative network G . Different from previous attempts, such as the convolutional neural network (CNN) models and the recurrent neural network (RNN) models, the generative adversarial architecture in GAN is benefit in following aspects. First, for data requirement, the adversarial architecture are trained in a semi-supervised way, which requires no additional label information. Second, for output results, the generative adversarial network is able to approximating intractable probabilities while other approaches tends to learn the average pattern [35].

In this study, considering the fact that the location of taxi hotspots is influenced by various types of dependencies, conditional information is applied in both generator G and discriminator D as an additional input layer [37]. With the conditional information, the generator G is able to specify the most probable outputs within the underlying distribution. The conditional structure applied in the proposed LSTM-CGAN model is illustrated in Fig. 1. As Fig. 1 shows, generator G builds a mapping function from a prior noise distribution p_z and the conditional information y to a data space, termed as $G(z|y)$. At the same time, the output $D(x|y_x)$ from discriminator D scales the probability that input data x comes from true data rather than G . Both G and D are trained simultaneously. The parameters in generator G are adjusted to fool the D , while the parameters in D are estimated to maximize the capability of identifying both the true data and samples output from G . In other words, generator G and discriminator D play a two-player min-max game, where G is updated to maximize the similarity between random samples and true data.

Furthermore, considering the temporal dependencies of taxi hotspots, two LSTM neural networks are utilized in the G and D , respectively. LSTM neural network, introduced as a special architecture [38], is benefit for dealing with long-term memory in temporal issues [25], [29]. A LSTM cell maps the input vector sequence $\{x, y_x\}$ to an output vector sequence h_t by T_0 iterations. The LSTM cell consists of an input layer, a memory block layer, and an output layer. The memory block contains three types of gating units to control information flow, which are the forget gate f_t , input gate i_t , and output gate o_t . Let c_t and c'_t

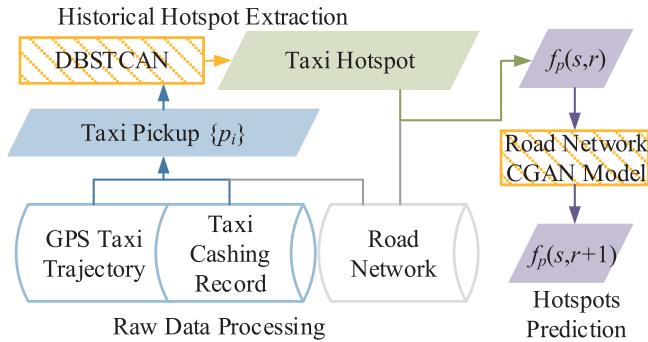


Fig. 2. Framework overview for the spatiotemporal taxi hotspot extraction and prediction.

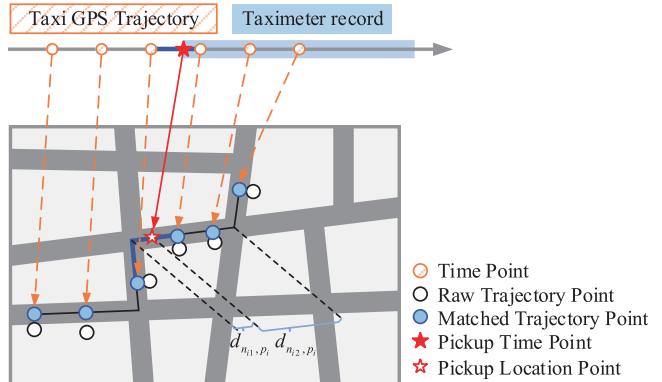


Fig. 3. Shortest path based linear interpolation located taxi pickup.

be the memory cell vector and the candidate value, respectively. All the information is updated as below:

$$f_t = \sigma(W_f[h_{t-1}, \{x, y_x\}_t] + b_f) \quad (1)$$

$$i_t = \sigma(W_i[h_{t-1}, \{x, y_x\}_t] + b_i) \quad (2)$$

$$c_t = \tanh(W_c[h_{t-1}, \{x, y_x\}_t] + b_c) \quad (3)$$

$$c_t = f_t * c_{t-1} + i_t * c_t \quad (4)$$

$$o_t = \sigma(W_o[h_{t-1}, \{x, y_x\}_t] + b_o) \quad (5)$$

$$h_t = o_t * \tanh(c_t) \quad (6)$$

where W_* and b_* imply the weight parameters and bias parameters in each operators, respectively; function $\sigma(\cdot)$ implies the standard logistics sigmoid function defined as:

$$\sigma(x) = \frac{1}{1 + \exp(-x)} \quad (7)$$

and function $\tanh(\cdot)$ is the standard hyperbolic tangent function defined as:

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \quad (8)$$

III. INTEGRATION FRAMEWORK OF EXTRACTION AND PREDICTION

In this section, we present the integration framework of extraction and prediction for taxi hotspot, including raw data processing, DBSTCAN-based taxi hotspot extracting, and LSTM-CGAN based hotspot predicting. An overview of the framework is illustrated in Fig. 2.

A. Raw Data Processing

The raw data includes GPS taxi trajectory, taximeter data, and the road network information. The GPS taxi trajectory records capture the following attributes: vehicle identification (ID), time stamp, longitude, latitude, and vehicle speed; the taximeter data has vehicle ID, pickup time, travel time, and income. For each taxi's trajectory data, we have the GPS location data for every 10 to 12 seconds. Considering that the average speed in urban area is usually no more than 80 km/hr, the distance between two subsequent GPS points is less than 250 m. The two GPS

points are most likely to be on the same road link or two adjacent connected links. This feature is applied as a rule in eliminating interference terms in trajectory map-matching. By combining the GPS taxi trajectory, taximeter record, and the road network information, we locate the pickup location onto the road network through a shortest path-based linear interpolation. Following steps are taken to implement the shortest path-based linear interpolation, as shown in Fig. 3.

- 1) For each taxi trajectory point, select a set of links that fall within a tolerance scope, which is determined by the associated GPS devices, using the vertical distance. Without loss of generality, 15 m is used in the present study;
- 2) If there exist several possible links associated with one GPS point, the adjacent links determined by the neighboring GPS points help select the most possible one. Then the taxi trajectory point (white point) is matched to nearest point on the most possible link, as the blue point shows;
- 3) Calculate the average speed for each spatiotemporal unit $v_{(s,r)}$;
- 4) The shortest path is calculated between the two locations based on their distance associated with a pickup time record (red star). More specifically, based on the pickup time point, the nearest two taxi trajectory points, i.e., before and after the pickup time point, are selected. As mentioned above, due to the high data frequency, two neighboring taxi trajectory points are usually on the same link or two adjacent links. The shortest path (blue line) connecting the two points are easily obtained.
- 5) Locate the pickup location on the shortest path linearly according to the time ratio.

Each pickup point, as a data object, is associated with the location information and the pickup time. At the same time, the spatiotemporal unit where the pickup point locates, and the distances to the two ends of recent link, as d_{n1,p_i} and d_{n2,p_i} shown in Fig. 3 are also recorded as attributes of the data object. It is worthwhile to mention that there exist errors in the proposed trajectory map-matching. In this study, more than 2 million data records were used, and there could be a very small proportion of pickup locations are mismatched. All the pickup locations are aggregated into the area/region mobility demand

Algorithm 1: Checking ε -Reachable Between p_i and p_j .

Input: p_i, p_j , threshold τ , shortest path matrix L^*

Output: ε -reachable indicator

```

1: if  $abs(p_i - p_j) > \tau$  then
2:   return 0
3: else
4:    $d_{p_i, p_j} \leftarrow \infty$ 
5:   for each  $n_{i,k}$  do
6:     for each  $n_{j,k}$  do
7:        $d_{p_i, p_j} \leftarrow \min\{d_{p_i, p_j}, d_{n_{i,k}, p_i} + l_{n_{i,k}, n_{j,k}} + d_{n_{j,k}, p_j}\}$ 
8:     end for
9:   end for
10: end if
11:  $l(p_i, p_j) \leftarrow \sqrt{d_{p_i, p_j}^2 + (v_{(s,r)}t_i - v_{(s,r)}t_j)^2}$ 
12: if  $l(p_i, p_j) > \varepsilon$  then
13:   return 0
14: else
15:   return 1
16: end if

```

information through the DBSTCAN clustering algorithm and LSTM-CGAN model investigate the underlying spatiotemporal distribution patterns of taxi demand. Therefore, the mismatching issues will not influence overall taxi demand distribution patterns significantly.

B. DBSTCAN-Based Taxi Hotspot Extracting

After all the pickup objects are located, we then apply the DBSTCAN algorithm to all these objects. Targeted distance $l(p_i, p_j)$ for any two pickup objects p_i and p_j is obtained following the pseudo-code (line 1 - line 10) shown in Algorithm 1. Considering the huge amount of pickup objects in the dataset, two approaches are applied to simplify the computations in spatiotemporal equivalent distance.

- Difference between the longitudes or the latitudes of the pickup objects larger than a pre-determined threshold τ is used to roughly eliminate most object pairs (line 1 - line 3). It shall be noted that the threshold to eliminate object pairs is determined subject to the neighborhood radius. For example, considering the latitude of the study area, a difference of 0.005 degree in longitudes or in latitudes indicates that the Euclidean distance between the two objects is larger than 500 m. In other words, the shortest path between the two objects is obviously larger than 500 m. If a neighborhood radius of 500 m is assumed, 0.005 degree can be roughly used as a threshold to determine a noise object.
- A novel shortest path calculation process is applied to simplify the distance computation at the expense of storage space (line 5 - line 9). In order to implement the DBSTCAN, the shortest path distance is required for every object pairs, which indicates huge amount of calculation. It is easy to understand that a number of taxi pickup objects located in the same links. Accordingly, instead of calculating the shortest path for each pickup object pair, a shortest path

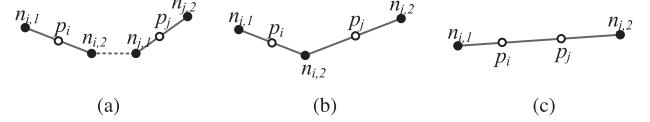


Fig. 4. Typical taxi pickup locations: (a) Type I; (b) Type II; (c) Type III.

matrix connecting all the intersections, termed as $L^* = \{l_{n_1, n_2}\}$, are obtained via the Floyd-Warshall algorithm and stored as known information. The shortest path connecting the two pickup objects is the minimal summation of the distances between pickup object p_i and the link end $n_{i,1}$, i.e., $d_{n_{i,1}, p_i}$, and the shortest path between the two link ends, i.e., $l_{n_{i,1}, n_{j,1}}$, which can be obtained in L^* directly. Possible positional relationship for taxi pickup pair are presented in Fig. 4. The signs for each distance value at line 7 in Algorithm 1 are optional considered for the relative locations for the taxi pickups and link ends.

Finally, targeted distance is obtained by combining the length of shortest path and equivalent distance of time interval between two pickups. Core and border objects are identified according to Definition 2. In preparing the data set for hotspots prediction, the core objects and border objects are then matched into spatiotemporal unit, i.e., the hotspot indicator $f_p(s, r)$. More specifically, the spatiotemporal units where the core objects locates, are assigned as 1, otherwise 0.

C. LSTM-CGAN Based-Hotspot Predicting

As mentioned above, two LSTM neural networks are utilized in the proposed CCAN structures, namely the LSTM-CGAN model. In the sample layout, shown in Fig. 1, a generator G consists of seven layers, which are one random generation layer and six LSTM layers; and a discriminator D also consists of four LSTM layers, one full connected layer, and one softmax layer. In this study, different cell numbers in a single LSTM layer were tested. In order to prevent over-fitting, we further apply the dropout method with a probability of 0.7 at each LSTM layer, which means that 30% cells are randomly ignore in each layer. The input of the D is designed to include a sequence of historical hotspot $x_r = \{f_p(s, r) | s \in S\}$ for each temporal unit and the corresponding conditional information $y_{x_r} = \{\{f_p(s, r-1) | s \in S\}, f_t(r)\}$, i.e., the taxi hotspot information in previous temporal unit and the time of day information. It is noted that time of day $f_t(r)$ is represented by the hour value for a temporal unit. Accordingly, the input shape of the discriminator D is $(11153 + 11153 + 1) \times 1$, i.e., a vector of 22307 elements. In this study, the time-step for look-back time window equals 6. With the input information, the output of each LSTM cell is a list of value between 0 and 1. The full connected layer together with the softmax function maps the output of the last LSTM layer to a value between 0 and 1, representing the probability that the input data is from real data. In the generator G , a sequence randomly samples are generated from a uniform distribution scaled between 0 and 1. The size of the random generated sample is just the same as vector x_r . Then, the uniform distributed samples, together with the corresponding conditional

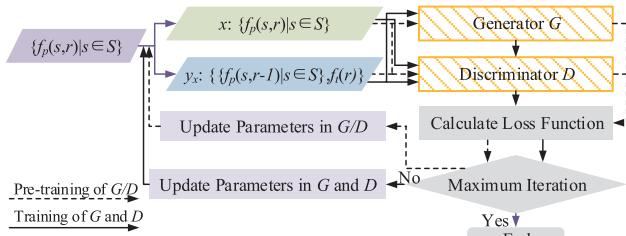


Fig. 5. Pre-training and training for the propose LSTM-CGAN model.

information, are imported into the six LSTM layers. As a result, the input shape of the generator G is also 22307×1 .

It has been universally acknowledged that training GAN/CGAN is hard [34], [37], [39]. In order to enhance the training process, pre-training procedure is applied to the generator G and discriminator D , as shown in Fig. 5. In the pre-training step, the parameters in G and D are updated to minimize the following loss functions, respectively.

$$l_G^P = E_{z \sim p_z}[x_r - G(z|y_{x_r})] \quad (9)$$

$$l_D^P = E_{x \sim p_{data}}[1 - D(x|y_x)] \quad (10)$$

When reaching the maximum iteration count, we stop the pre-training step and go to the training step. As indicated in previous section, D and G play the min-max game, which improves the capability for G in generating realistic samples. In updating the generator G and the discriminator D , we minimize the following loss functions for G and D , simultaneously.

$$l_G = E_{z \sim p_z}[\log(1 - D(G(z|y_x)|y_x))] \quad (11)$$

$$\begin{aligned} l_D = & E_{x \sim p_{data}}[\log(1 - D(x|y_x))] \\ & + E_{z \sim p_z}[\log D(G(z|y_x)|y_x)] \end{aligned} \quad (12)$$

As shown in Fig. 5, both the pre-training and training processes start from sampling mini-batch of examples from database. In this study the batch size is set to 36 as a tradeoff between training time per epoch and quality of model, as well as the required memory size [40]. The number of training epochs was set to 100 and 400 in pre-training and training process, respectively. Each example in the mini-batch consisted of taxi hotspot indicator set $x_r = \{f_p(s, r) | s \in S\}$ and the conditional information set y_{x_r} . The generator G is pre-trained comparing the outputs and the field collected data x_r following Eq. (9), while the discriminator D is pre-trained using the two sets with an un-trained generator G following Eq. (10) for pre-specified iteration threshold. Then for training process, the conditional information y_{x_r} is imported to both the generator G and discriminator D , while x_r is imported to discriminator. And parameters in the two networks are updated simultaneously following Eq. (11) and Eq. (12). It shall be noted that convergence on training dataset and prediction performance on testing datasets are different. In this study, we mainly focus on the prediction performance, and are simply training the model as long as necessary, which ensures the LSTM-CGAN model captures all the possible spatiotemporal patterns hiding in the training dataset.

After the CGAN is well trained, taxi hotspots are generated by the generator G with conditional information. It is noted that multiple samples will be generated and accordingly, multiple hotspot predictions are calculated. The average value of the multiple hotspot predictions is used as the predicted taxi hotspots value. The predicted taxi hotspots are then obtained by applying the following classification function to the model outputs of the last time-step. A pre-specified threshold 0.5 is used here.

$$f_p^*(s, r+1) = \begin{cases} 1 & f_p(s, r+1) \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

IV. EXPERIMENTAL RESULTS

This section presents performance of the proposed taxi hotspot extraction and prediction framework using field collected taxi trajectory data. Moreover, the proposed framework was compared with four other prediction methods, including the moving average (MA) method, the autoregressive integrated moving average (ARIMA) method, the LSTM neural network model, and the LSTM-GAN model.

A. Data Description

The taxi trajectory data and taximeter data used in this paper was collected from the Beijing taxi companies. The number of referred taxis was 68310. As shown in the annual city report of Beijing, the total number of operating taxis was around 68000, which indicated that the authors obtained all the accessible taxi data during the studying period. The studying area located in downtown Beijing, China, from 116.19° E to 116.39° E in longitude, and from 39.93° N to 40.13° N in latitude. The total number of pick-up records in studying area was 2640826. Road network information was downloaded from Open Street Map. Studying period in this study was 2 weeks (from Nov. 2nd, 2015 to Nov. 15th, 2015). More specifically, data from Nov. 2nd to Nov. 12th was used as the training set, data in Nov. 13th was used as a validating set, and the other two days were used for testing. It was noted that a two-week dataset may not be large enough to cover various traffic patterns/situations, such as weekend patterns, emergency situations, or adverse weather conditions. The major purpose of this study was to develop the extraction and prediction framework for taxi hotspot, which combined the DBSTCAN algorithm and the LSTM-CGAN model for the first time, and examined its feasibility and predictive performance under prevailing conditions. Two-week traffic data under prevailing conditions is sufficient to demonstrate major, regular traffic operation patterns for the proposed method to capture the spatiotemporal features in terms of preliminary testing purposes.

Fig. 6 showed the spatial distribution of taxi pickups in different time periods for both weekday, i.e., Fig. 6(a), (c), (e), and (g), and weekends, i.e., Fig. 6(b), (d), (f), (h). It was clear that the number of pickup objects during weekdays was significantly higher than the number during weekends, as Fig. 6(a), (c), (e), and (g) were darker than Fig. 6(b), (d), (f), (h). And it can also be observed from Fig. 6, which showed a time-varying pattern of the dense sections for taxi pickup as expected. The dynamic evolution of taxi hotspots was trackable by coherent illustrating the

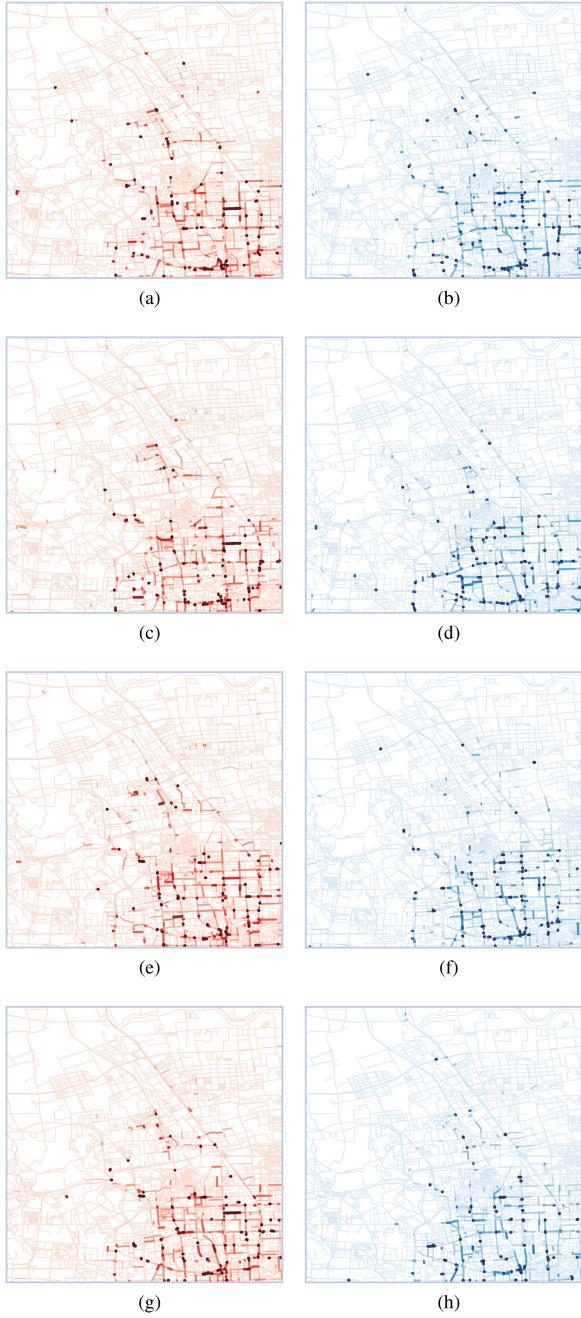


Fig. 6. Spatial distribution of taxi pickups at different time periods: (a) 12:00–12:10, Nov. 2nd; (b) 12:00–12:10, Nov. 7th; (c) 12:10–12:20, Nov. 2nd; (d) 12:10–12:20, Nov. 7th; (e) 12:20–12:30, Nov. 2nd; (f) 12:20–12:30, Nov. 7th; (g) 12:30–12:40, Nov. 2nd; (h) 12:30–12:40, Nov. 7th.

spatial distribution of the taxi pickups. The time-varying pattern motivated adding a temporal index as a conditional information in the proposed LSTM-CGAN model. As mentioned above, the temporal index was involved for different hour.

B. Extracting Historical Taxi Hotspots

In implementing the DBSTCAN algorithm, the minimum required object count m_p was roughly set as 4, which was recommended in previous studies [13]. The neighborhood radius ε was set as 1000 m, which also presented a spatiotemporal

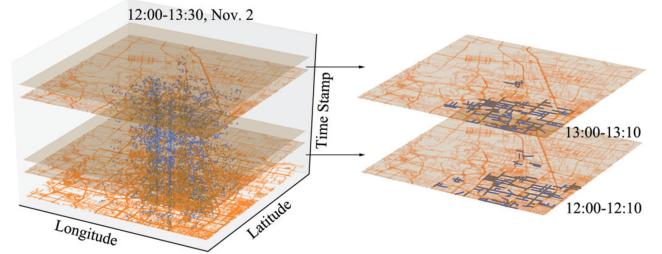


Fig. 7. Taxi hotspots extracting using DBSTCAN algorithm.

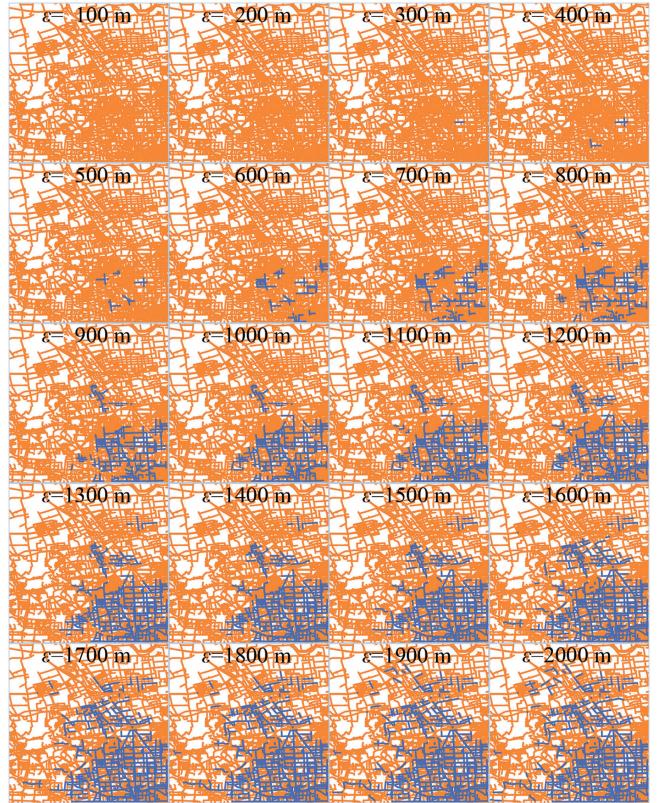


Fig. 8. Spatial distribution of taxi hotspots during 12:00–12:10, Nov. 2nd with different neighborhood radius.

radius of taxi-passenger searching behavior. Fig. 7 illustrated the outputs of the proposed DBSTCAN algorithm. As illustrated in Fig. 7, all the core objects located between 12:00–13:30, Nov. 2nd were marked using blue points in the time-space coordinate. And then taxi hotspots were extracted according to Definition 4. It was noted that the extracted taxi hotspots were aggregated for every 10 min in this study, as shown on the right side of Fig. 7. Taxi hotspots in the two periods were marked in blue, respectively.

Moreover, we further tested the relationship between the neighborhood radius ε and the extraction results of taxi hotspots. Taking the extracted taxi hotspots during 12:00–12:10, Nov. 2nd as an example, Fig. 8 presented the spatial distribution of those taxi hotspots, *w.r.t.*, varying neighborhood radius ε . More road sections within this time period were extracted as hotspots as the increased neighborhood radius ε . It was easy to understand

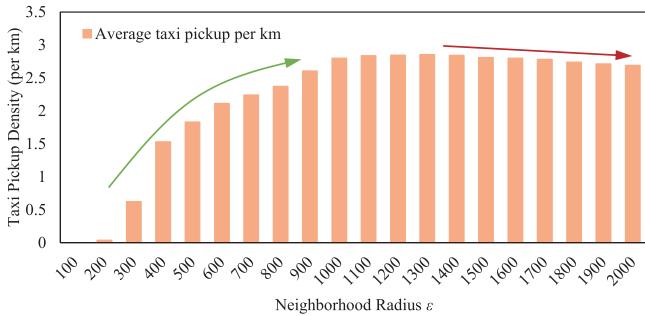


Fig. 9. Impacts of different neighborhood radius ε on the extracted taxi hotspots.

that, eventually, most of the road sections may be involved into the extraction results with a large enough ε , which was out of the scope for hotspot extraction. Fig. 9 demonstrated the average pickup density within the extracted taxi hotspots, *w.r.t.*, different neighborhood radii. It was found that the average taxi pickup density increased slower as the neighborhood radius larger than 1000 m, and slightly decreased when the neighborhood radius grew up to 1500 m, which indicated the fact that with the increasing neighborhood radius, a number of less dense sections were involved into the extraction results. Accordingly, we applied the extraction results of neighborhood radius $\varepsilon = 1000$ m. And the impacts of the neighborhood radii on the final prediction result were also presented in the next section.

C. Performance Evaluation of Taxi Hotspots Prediction Using LSTM-CGAN

Two quantitative methods were selected to evaluate the performance of the proposed LSTM-CGAN model, *i.e.*, the false identification test (FIT), and the section consistency test (SCT) [41]. They were implemented as follows:

- FIT: the false identification test considers the false negative and false positive associated with the prediction results. The false negative rate (FIT-N) represents the proportion of the truly hotspot sections being mistakenly predicted as non-hotspot sections, while the false positive rate (FIT-P) represents the proportion of the non-hotspot sections being mistakenly identified as hotspot sections. The FIT-N and FIT-P were given as the proportion of the total length of false identified non-hotspot sections in the true hotspot sections and the proportion of the total length of false identified hotspot sections in the true non-hotspot sections, respectively.
- SCT: the section consistency test is proposed as a supplement to the FIT test. It represents the ratio of the taxi pickup density in predicted taxi hotspot sections to the density in truly hotspot sections.

The proposed LSTM-CGAN model for taxi hotspots prediction was trained based on data collected from Nov. 2nd to Nov. 12th, and was validated using data collected in Nov. 13th. Considering the pre-specified batch size and epoch size, it was 44 iterations per epoch, and 22000 iterations in total. Results

TABLE II
VALIDATION RESULTS WITH DIFFERENT NUMBER OF LSTM LAYERS IN GENERATOR AND DISCRIMINATOR

	Index	# of LSTM layers in G				Avg.
		2	4	6	8	
# of LSTM layers in D	FIT-N	41.0%	41.2%	41.8%	41.6%	41.4%
	FIT-P	0.54%	0.54%	0.53%	0.53%	0.53%
	SCT	74.2%	74.8%	74.9%	74.7%	74.6%
4	FIT-N	41.3%	39.8%	41.8%	41.5%	41.1%
	FIT-P	0.53%	0.53%	0.53%	0.54%	0.53%
	SCT	74.3%	75.2%	75.1%	75.3%	75.0%
6	FIT-N	41.7%	41.5%	41.4%	41.9%	41.6%
	FIT-P	0.54%	0.53%	0.53%	0.53%	0.53%
	SCT	74.4%	75.2%	75.1%	74.8%	74.9%
8	FIT-N	41.3%	41.5%	41.4%	41.4%	41.4%
	FIT-P	0.54%	0.54%	0.54%	0.54%	0.54%
	SCT	74.3%	74.9%	74.9%	75.0%	74.8%
Avg.	FIT-N	41.3%	41.0%	41.6%	41.6%	
	FIT-P	0.54%	0.53%	0.53%	0.54%	
	SCT	74.3%	75.0%	75.0%	75.0%	

of performance evaluation with different model layouts, *i.e.*, well trained LSTM-CGAN models with different numbers of LSTM layers in generator G and discriminator D , were listed in Table II. Note that for each LSTM layer, the size of hidden cells for each layout was determined by assigning values from the set of 256, 512, 1024, and 2048, for obtaining the optimal hidden cell size [26], [29].

As shown in Table II, 16 different layouts of the proposed LSTM-CGAN model were trained and the validation results were evaluated using the three criterion. In general, the proposed models in different layouts presented quite similar performance. More specifically, the false negative rate ranged from 39.8% to 41.9%; the false positive rate ranged from 0.53% to 0.54%; and the section consistency rate ranged from 74.2% to 75.3%. It was found that the prediction performance was improved as the number of LSTM layers increasing from 2 to 4 for both the generator G and the discriminator D , and ran back slightly as the number of LSTM layers increasing up to 8. Considering the increasing complexity of model structures, the ability of proposed LSTM-CGAN model for capturing underlying relationship among imported information and prediction was enhanced. And the possible explanation for the declines was the overfitting problem. As indicated in Table II, the proposed LSTM-CGAN model with 4 LSTM layers in both generator G and discriminator D performed best.

Fig. 10 presented the distribution of predicted hotspot sections and corresponding taxi pickups from 12:00 to 12:40, Nov. 13th. In the up-bound graphs of Fig. 10, four colors are used to differentiate different prediction results. More specifically, the orange and blue segments are correctly predicted non-hotspots and hotspots, respectively; the red segments are the non-hotspot sections being mistakenly predicted as hotspot sections; and the green segments are the hotspot sections being mistakenly predicted as non-hotspot sections. In the down-bound graphs of Fig. 10, the shaping indicates the concentration ratio of taxi pickups. It was found that the proposed LSTM-CGAN model captured the spatiotemporal relationships among the historical hotspot information successfully. And the prediction results showed low proportion of red segments, *i.e.*, the FIT-P. And when come to the green segments, which indicated the FIT-N,

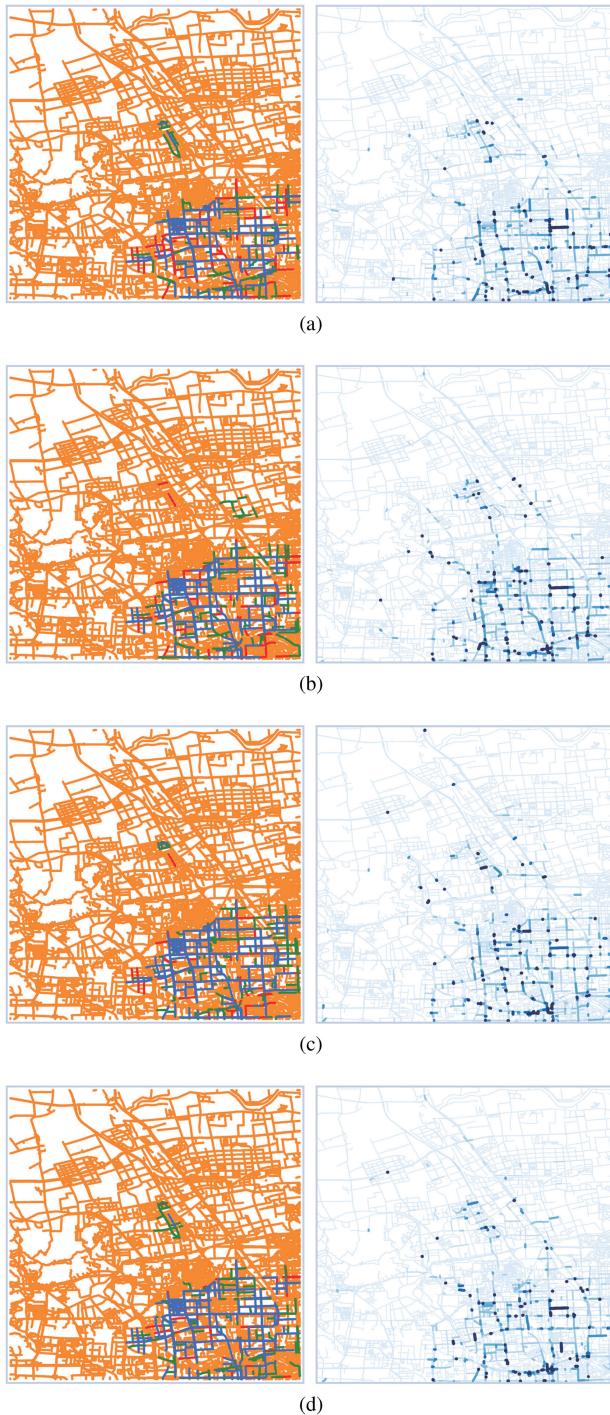


Fig. 10. Prediction results of taxi hotspot sections and corresponding taxi pickups: (a) 12:00–12:10, Nov. 13th; (b) 12:10–12:20, Nov. 13th; (c) 12:20–12:30, Nov. 13th; (d) 12:30–12:40, Nov. 13th.

it was found that the taxi pickups in the green segments were not too much. As a result, the actual taxi pickups covered in predicted segments were more than 75%, shown in Table II.

D. Model Comparison

Four prediction approaches compared in this study were briefly described as follows:

TABLE III
COMPARISON OF PREDICTIVE PERFORMANCE USING TESTING DATASET

Model	Index	N	Min	Max	Mean	Std.
MA	FIT-N	288	0.00%	100.0%	63.50%	21.79%
	FIT-P	288	0.00%	9.77%	4.23%	6.61%
	SCT	288	0.00%	100.00%	41.78%	33.83%
ARIMA	FIT-N	288	0.00%	100.00%	53.91%	13.81%
	FIT-P	288	4.15%	5.08%	4.78%	0.34%
	SCT	288	1.91%	100.00%	48.84%	20.23%
LSTM	FIT-N	288	0.00%	100.00%	43.96%	15.19%
	FIT-P	288	3.90%	6.45%	5.20%	0.66%
	SCT	288	1.76%	100.00%	47.20%	20.31%
LSTM-GAN	FIT-N	288	0.00%	100.00%	47.77%	16.11%
	FIT-P	288	3.75%	5.34%	4.82%	0.49%
	SCT	288	1.75%	100.0%	46.86%	20.32%
LSTM-CGAN	FIT-N	288	0.00%	100.00%	39.93%	13.80%
	FIT-P	288	0.02%	1.02%	0.56%	0.29%
	SCT	288	7.30%	100.00%	73.43%	29.73%

- Moving Average (MA): the MA method was one of the most disseminated approaches in time-series analysis, which calculates predictions using the mean value of nearest historical records. In this study, the time window equaled six time-steps, i.e., the taxi hotspot indicators in spatiotemporal units $(s, r - 6)$ to $(s, r - 1)$ were used to predict the taxi hotspot indicator in the spatiotemporal unit (s, r) . It should be noted that Eq. (13) was applied to adjust the model outputs.
- Auto-regressive Integrated Moving Average (ARIMA): in an ARIMA (p, d, q) model, the integer parameter p and q are referred as the orders of the autoregressive term and moving average term, respectively, while the parameter d represents the d^{th} order difference from the original data series, which aims to remove the trend from the data series [42]. In this study, parameters were optimized using the auto-optimal function in forecast package with R-3.4.4.
- LSTM neural network: the LSTM neural network was also applied for taxi hotspot prediction. The LSTM model conducted here was formulated in the same structure as the generative network G in the proposed LSTM-CGAN model except for the random generation layer, i.e., four LSTM layers and one full connected layer. More specifically. Conditional information $\{y_x\}$ was used as an input considering 6 time-steps for look-back time window, while the output was adjusted using Eq. (13).
- LSTM-GAN: the LSTM-GAN model was also conducted in the test. The main difference between the proposed LSTM-CGAN model and the LSTM-GAN model applied here located on the existing of conditional information $\{y_x\}$.

Note that the ARIMA model was calibrated for each single road sections and the predicted hotspot indicator for each spatiotemporal unit was obtained separately, while the LSTM, and the LSTM-GAN models were applied to the whole network. Table III compared the evaluation results of the proposed LSTM-CGAN model and the other four approaches. Note that the LSTM-CGAN model refers to the CGAN layout with four LSTM layers in both G and D . As shown in Table III, each test was applied to each time period using testing dataset.

As shown in Table III, the proposed LSTM-CGAN model outperformed the other approaches in all the three performance

TABLE IV
STATISTICAL DESCRIPTION FOR TIME-VARYING HOTSPOT COUNTS

Time Period	N	Min	Mean	Max	Std
0:00-2:00	168	0 (0%)	19.74 (0.177%)	134 (1.201%)	20.63 (0.185%)
4:00-6:00	168	0 (0%)	3.15 (0.028%)	18 (0.161%)	4.50 (0.040%)
8:00-10:00	168	0 (0%)	1.42 (0.013%)	13 (0.117%)	2.70 (0.024%)
12:00-14:00	168	0 (0%)	50.65 (0.454%)	193 (1.730%)	47.25 (0.424%)
16:00-18:00	168	52 (0.466%)	271.16 (2.431%)	519 (4.653%)	120.72 (1.082%)
20:00-22:00	168	125 (1.121%)	337.67 (3.028%)	494 (4.429%)	90.59 (0.812%)
12:00-13:59	168	214 (1.919%)	372.44 (3.339%)	586 (5.254%)	101.99 (0.914%)
14:00-15:59	168	176 (1.578%)	398.21 (3.570%)	589 (5.281%)	110.57 (0.991%)
16:00-17:59	168	161 (1.444%)	312.12 (2.799%)	491 (4.402%)	83.43 (0.748%)
18:00-19:59	168	110 (0.986%)	239.75 (2.150%)	353 (3.165%)	49.95 (0.448%)
20:00-21:59	168	102 (0.915%)	248.57 (2.229%)	412 (3.694%)	66.67 (0.598%)
22:00-23:59	168	0 (0%)	139.42 (2.977%)	332 (0.637%)	71.09 (0.637%)

indexes. It was found that the three neural network based approaches performed much better than the other two time-series approaches, i.e., the MA and the ARIMA approaches. The two time-series approaches took no considering about the spatial relationship among sections. It would be difficult for the MA and the ARIMA approaches to capture the time-varying hotspots, resulting in worse performance in the tests. When we came to the LSTM approach and the LSTM-GAN approach, it was found that the LSTM achieved better performance in the FIT-N and the SCT tests than the LSTM-GAN. The LSTM model applied the historical taxi hotspot information to predict the hotspots directly, while the LSTM-GAN model tried to capture the spatiotemporal relationship among the taxi hotspots and the randomly generated the prediction results by generator G . The LSTM-CGAN model combined the benefits of the conditional information and the adversarial training procedure together, and significantly improved the predictive performance in all the tests. The proposed LSTM-CGAN reduced the FIT-P from 5.20% and 4.82% to 0.56%; the FIT-N was reduced from 43.96% and 47.77% to 39.93%; the SCT was improved from 47.20% and 46.86% to 73.43%, which showed that the proposed LSTM-CGAN model captured the ability to predict dense sections of taxi pickups.

Moreover, Table IV presented a statistical description for the hotspot count of each time step within different time periods. The value in parentheses indicates the hotspot counts as a percentage of total spatial unit counts. It was found that, the hotspot counts not only change with time, i.e., the average number of hotspots ranges from 1.42 to 398.21, but also vary a lot within each time period, i.e., the standard deviation for each time period is close to the average value. Accordingly, it is clear to see that the dataset is highly unbalanced and the taxi hotspots change greatly in both spatial and temporal dimensions.

In order to better understand the superiority of the proposed method against the reference methods, especially for the

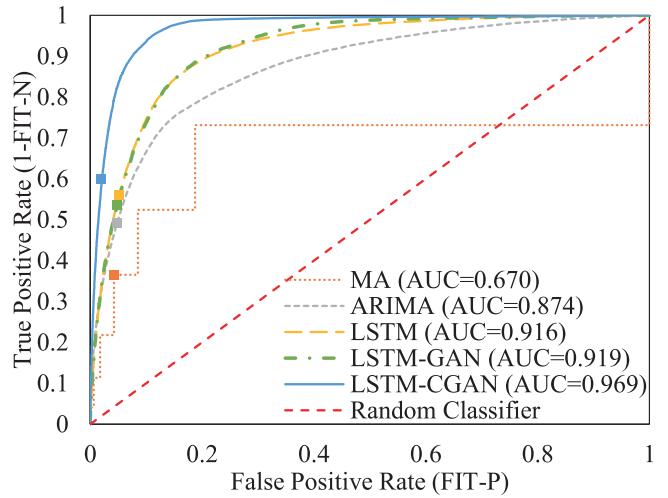


Fig. 11. ROC curves for the prediction performance considering various thresholds.

unbalanced dataset, the receiver operating characteristic (ROC) curve is employed, and the performance of difference approaches is measured by the area under the curve (AUC). More specifically, the ROC curve is a graphical plot that illustrates the prediction performance of the proposed model as the discrimination threshold is varied. It is created by plotting the true positive rate ($1 - \text{FIT-N}$), against the false positive rate (FIT-P), and the area under the receiver operating characteristic (AUROC) [43]. Fig. 11 illustrated the ROC curves for the prediction performance of the five methods. Note that the prediction performance using 0.5 as the threshold is labeled with a solid block on each curve. As shown in Fig. 11, almost all the five methods perform better than the random classifier, except for the MA method when the threshold is very low. It was found that the proposed LSTM-CGAN outperformed all the other methods, followed by the LSTM-GAN model, the LSTM model, the ARIMA model, and the MA method.

V. CONCLUSION

This study investigated how the integration of clustering model and deep learning approach contributes to extracting and predicting network-wide taxi hotspots by leveraging multi-source datasets. The network-wide taxi hotspot extraction and prediction framework is developed by integrating a density-based spatiotemporal clustering algorithm with noise (DBST-CAN) and a conditional generative adversarial network with long short-term memory structure (LSTM-CGAN). The DBST-CAN algorithm was applied to capture the spatiotemporal correlation of taxi pickups by applying the spatiotemporal distance. The LSTM-CGAN was proposed for predicting the time-varying taxi hotspot sections. The predictive performance of the model with different model layouts, i.e., different numbers of LSTM layers in G and D , were compared using the validation dataset. It was found that all those models provided comparable results, and the ability to capture the spatiotemporal correlation among historical hotspot information was enhanced by increasing the

number of LSTM layers. The proposed LSTM-CGAN model with 4 LSTM layers in both generator G and discriminator D performed best. However, as illustrated by the performance indexes, the proposed models suffered from the overfitting problem. The possible solutions can be to enlarge the training dataset size, and apply the techniques of the ResNet architecture and regularization [44].

Moreover, two commonly-used models for time-series traffic prediction (MA and ARIMA), and two state-of-the-art machine learning models for network-wide taxi hotspot prediction (LSTM and LSTM-GAN) were selected as benchmark methods to compare with the proposed LSTM-CGAN in two quantitative tests (FIT and SCT). The ROC analysis was then conducted to further investigate the prediction performance of the proposed LSTM-CGAN models with varying discrimination thresholds. The comparative analyses indicated that the proposed LSTM-CGAN outperformed the benchmark approaches in terms of lower FIT, higher SCT and AUC. More specifically, with the proposed LSTM-CGAN, the FIT-P was reduced from 5.20% to 0.56%; the FIT-N was reduced from 63.50% to 39.93%; the SCT was improved from 41.78% to 73.43%; and the AUC was improved from 0.670 to 0.969. Additionally, the comparison results revealed that the combination of the adversarial training procedure and the conditional information, as well as the LSTM technique, improved the model performance significantly. It was necessary to note that the adversarial training procedure enabled the generator G in the proposed LSTM-CGAN model to gain better understanding of the spatiotemporal distribution of taxi hotspots with conditional information, i.e., hotspot locations in previous time periods and the time of day information.

The proposed extraction and prediction framework provided an integrated approach to better understand the spatiotemporal correlation among taxi hotspot sections. The potential engineering applications are mainly twofold. Firstly, for city planners, the predicted taxi hotspot results can help traffic management agencies to provide taxi-friendly facilities in network-wide infrastructure. Secondly, for taxi managers, the time-vary prediction results of the proposed LSTM-CGAN model reveal the taxi hotspots dynamically. The predicted hotspot information can be provided to selected cruising taxis in neighborhood, which is particularly important during peak hours. Moreover, the proposed prediction model can be enhanced by adding more conditional information, such as drivers route choice and trip destination information, which can incorporate the predicted taxi hotspots into the current shared mobility applications.

Even though the integrated framework has exhibited great potential to locate network-wide taxi hotspots accurately, several limitations are still needed to be addressed in the further study. Firstly, only two weeks data was applied in training, validating and testing the proposed model. More data are desirable to cover various traffic patterns/situations, such as weekend patterns, emergency situations, or adverse weather conditions. Especially for emergency situations and adverse weather conditions, the travel demands may reduce in some areas, but increase in other areas. Taxi hotspots may significantly change in spatiotemporal dimensions to reflect abnormal mobility demands under these conditions. These taxi hotspot changes are influenced by many

factors, such as storm types, magnitudes, durations, and coverages, as well as the infrastructure resilience degree under extreme climate and weather conditions. More data for longer time periods are necessary to test the stability of the proposed framework. The factors affecting taxi hotspots in spatiotemporal dimensions shall be carefully investigate in future research. Secondly, the proposed LSTM-CGAN model involves only the taxi pick-up hotspot information. For future research, we recommend more information from various data sources should be included. For instance, a comprehensive model with the pick-up information, network characteristics, and socioeconomic patterns would be pursued to improve the predictive accuracy. In that case, the impacts of the hyper-parameters from the DBSTCAN model on the predictive performance shall also be investigated. Furthermore, it is notable that real-time taxi demands requesting and processing are needed in applying the proposed extraction and prediction framework. Therefore, in-depth research addressing the information and communication issues for the proposed framework in practical application is of great interests and could be explored in future research.

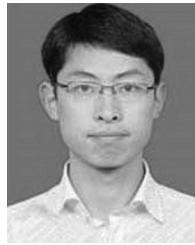
ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers for critically reading the earlier draft of the manuscript and suggesting substantial improvements.

REFERENCES

- [1] X. Kong, F. Xia, J. Wang, A. Rahim, and S. K. Das, "Time-location-relationship combined service recommendation based on taxi trajectory data," *IEEE Trans. Intell. Transportation Syst.*, vol. 13, no. 3, pp. 1202–1212, Jun. 2017.
- [2] R.-h. Hwang, Y.-l. Hsueh, and Y.-t. Chen, "An effective taxi recommender system based on a spatio-temporal factor analysis model," *Inf. Sci.*, vol. 314, pp. 28–40, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.ins.2015.03.068>
- [3] M. Zhang, J. Liu, Y. Liu, Z. Hu, and L. Yi, "Recommending pick-up points for taxi-drivers based on spatio-temporal clustering," in *Proc. 2nd Int. Conf. Cloud Green Comput. Recommending*, 2012, pp. 67–72.
- [4] J. Lee, I. Shin, and G.-l. Park, "Analysis of the passenger pick-up pattern for taxi location recommendation," in *Proc. 4th Int. Conf. Networked Comput. Adv. Inf. Manag. Anal.*, 2008, pp. 199–204.
- [5] H. W. Chang, Y. C. Tai, and J. Y. J. Hsu, "Context-aware taxi demand hotspots prediction," *Int. J. Bus. Intell. Data Mining*, vol. 5, no. 1, pp. 3–18, 2010.
- [6] N. Verma and N. Balyan, "PAM clustering based taxi hotspot detection for informed driving," in *Proc. 8th Int. Conf. Comput., Commun. Netw. Technol.*, 2017, pp. 1–7.
- [7] H.-W. Dong, W.-Y. Hsiao, L.-C. Yang, and Y.-H. Yang, "MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment," in *Thirty-Second AAAI Conf. Artif. Intell.*, 2018, pp. 34–41.
- [8] J. W. Powell, Y. Huang, F. Bastani, and M. Ji, "Towards reducing taxicab cruising time using spatio temporal profitability maps," *Advances in Spatial and Temporal Databases*, Berlin, Germany: Springer, 2011, pp. 242–260.
- [9] X. Wan and J. Wang, "DBH-CLUS: A hierarchical clustering method to identify pick-up/drop-off hotspots XueJin," in *Proc. 15th IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput. DBH-CLUS*; 2015, pp. 890–897.
- [10] J. Yuan, Y. Zheng, L. Zhang, X. Xie, and G. Sun, "Where to find my next passenger," in *Proc. 13th Int. Conf. Ubiquitous Comput.*, Beijing, 2011, pp. 109–118. [Online]. Available: <http://doi.acm.org/10.1145/2030112.2030128>
- [11] K. Niu, C. Cheng, C. Jielin, H. Zhang, and T. Zhou, "Real-time taxi-passenger prediction with L-CNN," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4122–4129, May 2019.

- [12] J. Tang, F. Liu, Y. Wang, and H. Wang, "Uncovering urban human mobility from large scale taxi GPS data," *Physica A*, vol. 438, pp. 140–153, 2015.
- [13] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.
- [14] D. Kumar, H. Wu, Y. Lu, S. Krishnaswamy, and M. Palaniswami, "Understanding urban mobility via taxi trip clustering," in *Proc. 17th IEEE Int. Conf. Mobile Data Manag.*, 2016, pp. 318–324.
- [15] E. Kourtzi, C. Christodoulou, L. Dimitriou, S. Christodoulou, and C. Antoniou, "Quantifying demand dynamics for supporting optimal taxi services strategies," *Transp. Res. Procedia*, vol. 22, pp. 675–684, 2017. [Online]. Available: <http://dx.doi.org/10.1016/j.trpro.2017.03.065>
- [16] Y. Liu, J. Liu, Z. Liao, M. Tang, and J. Chen, "Recommending a personalized sequence of pick-up points," *J. Comput. Sci.*, 2017. [Online]. Available: <http://dx.doi.org/10.1016/j.jocs.2017.05.004>
- [17] J. Ke, H. Zheng, H. Yang, and X. M. Chen, "Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach," *Transp. Res. Part C: Emerging Technol.*, vol. 85, no. October, pp. 591–608, 2017. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0968090X17302899>
- [18] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2016, pp. 1655–1661. [Online]. Available: <http://arxiv.org/abs/1610.00081>
- [19] X. Qian and S. V. Ukkusuri, "Spatial variation of the urban taxi ridership using GPS data," *Appl. Geography*, vol. 59, pp. 31–42, 2015. [Online]. Available: <http://dx.doi.org/10.1016/j.apgeog.2015.02.011>
- [20] Y. Shi, H. Liu, L. Gao, and G. Zhang, "Cellular particle swarm optimization," *Inf. Sci.*, vol. 181, no. 20, pp. 4460–4493, 2011. [Online]. Available: <http://dx.doi.org/10.1016/j.ins.2010.05.025>
- [21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [22] Y. Lv, Y. Duan, W. Kang, and F.-Y. Wang, "Traffic flow prediction with big data : A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2014.
- [23] A. Koeswiady, R. Soua, and F. Karray, "Improving traffic flow prediction with weather information in connected cars: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9508–9517, Dec. 2016.
- [24] D. Jo, B. Yu, H. Jeon, and K. Sohn, "Image-to-image learning to predict traffic speeds by considering area-wide spatio-temporal dependencies," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1188–1197, 2019.
- [25] Y. Liang, Z. Cui, Y. Tian, H. Chen, and Y. Wang, "A deep generative adversarial architecture for network-wide spatial-temporal traffic state estimation," in *Proc. Transp. Res. Board 97th Annu. Meet.*, Washington, D.C. USA, 2018, pp. 87–105.
- [26] J. Bao, P. Liu, and S. V. Ukkusuri, "A spatiotemporal deep learning approach for citywide short-term crash risk prediction with multi-source data," *Accident Anal. Prevention*, vol. 122, no. November 2018, pp. 239–254, 2019. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0001457518303877>
- [27] J. Bao, H. Yu, and J. Wu, "Short-term ffbs demand prediction with multi-source data in a hybrid deep learning framework," *IET Intell. Transport Syst.*, vol. 13, no. 9, pp. 1340–1347, 2019.
- [28] H. Yu *et al.*, "Taxi-based mobility demand formulation and prediction using conditional generative adversarial network-driven learning approaches," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3888–3899, 2019.
- [29] X. Ma, H. Yu, Y. Wang, and Y. Wang, "Large-scale transportation network congestion evolution prediction using deep learning theory," *PLoS ONE*, vol. 10, no. 3, pp. 1–17, 2015.
- [30] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transp. Res. Part C: Emerging Technol.*, vol. 43, pp. 3–19, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.trc.2014.01.005>
- [31] M. S. Dougherty and M. R. Cobbett, "Short-term inter-urban traffic forecasts using neural networks," *Int. J. Forecasting*, vol. 13, no. 1, pp. 21–31, 1997.
- [32] S. Clark, "Traffic prediction using multivariate nonparametric regression," *J. Transp. Eng.*, vol. 129, no. 2, pp. 161–168, 2003.
- [33] S. V. Kumar and L. Vanajakshi, "Short-term traffic flow prediction using seasonal ARIMA model with limited input data," *Eur. Transport Res. Rev.*, vol. 7, no. 3, pp. 1–9, 2015.
- [34] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [35] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 2255–2264.
- [36] Y. Lv, Y. Chen, L. Li, and F.-Y. Wang, "Generative adversarial networks for parallel transportation systems," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 3, pp. 4–10, Fall 2018.
- [37] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv: 1411.1784*.
- [38] S. Hochreiter and J. Urgen Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [39] B. Neyshabur, S. Bhojanapalli, and A. Chakrabarti, "Stabilizing GAN training with multiple random projections," 2017, *arXiv: 1705.07831*.
- [40] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, "On large batch training for deep learning: Generalization gap and sharp minima," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–16.
- [41] W. Cheng and S. Washington, "New criteria for evaluating methods of identifying hot spots," *Transp. Res. Rec.: J. Transp. Res. Board*, vol. 2083, pp. 76–85, 2008. [Online]. Available: <http://trrjournalonline.trb.org/doi/10.3141/2083-09>
- [42] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, "Linear nonstationary models," *Time Series Anal.*, pp. 93–136, 2013. [Online]. Available: <http://doi.wiley.com/10.1002/9781118619193.ch4>
- [43] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.
- [44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 770–778.



Hao Yu received the B.S and Ph.D. degrees from the School of Transportation, Southeast University, Nanjing, China, in 2000 and 2017, respectively. He is currently a Research Associate with the Department of Civil and Environmental Engineering, University of Hawaii at Mānoa, Honolulu, HI, USA. His research interests include large-scale traffic network modeling, big data on urban traffic management, highway safety analysis, and their practical engineering applications. He serves as member of the Transportation Research Board (TRB) Committee of Information System and Technology (ABJ50), and a member of the TRB Committee of Agricultural and Food Transportation (AT030).



Zhenning Li received the Ph.D. degree from the University of Hawaii at Mānoa, Honolulu, HI, USA. He is currently a Postdoctoral Researcher with the Department of Civil and Environmental Engineering, University of Macau, Macau, China. His research interests include traffic modeling in connected autonomous vehicles systems, big data on urban traffic management and spatiotemporal analysis on highway safety.



Guohui Zhang (Member, IEEE) received the Ph.D. degree from the University of Washington, Seattle, WA, USA, in 2000. He is currently an Associate Professor with the Department of Civil and Environmental Engineering, University of Hawaii at Mānoa, Honolulu, HI, USA. His research interests include transportation system operations and analysis, transportation data management and analysis, large-scale traffic simulation, geographic information system-based infrastructure asset management, and traffic detection systems. He serves as a member of the Transportation Research Board Committee on Information Systems and Technology (ABJ50), a member of NCHRP Project Panel on Work Zone Capacity Methods for the Highway Capacity Manual (D03107), a member of NCHRP Project Panel on Environmental Justice Analyses When Considering Toll Implementation or Rate Changes (D08100), and an Executive Board Member of the Chinese Overseas Transportation Association (COTA). He was the recipient of the Rising Star Award from the UW Smart Transportation Applications and Research Laboratory in 2008 and the Traffic Simulation Award from Planung Transport Verkehr (PTV) AG in 2009.



Pan Liu received the Ph.D. degree in civil engineering from the University of South Florida, Tampa, FL, USA, in 2006. He is currently a Professor with the Key Laboratory of Traffic Planning and Management, School of Transportation, Southeast University, Nanjing, China. He is serving as a Visiting Professor with the Institute of Road Traffic Management at the Ministry of Public Security of the People's Republic of China. His research interests include traffic design, traffic safety, and intelligent transportation systems. He is a member of the expert committee of the National Associate of the Intelligent Transportation Systems, and the vice Chairman of the Young Scientists Committee of the National Transportation Association. He is the recipient of the Outstanding Youth Science Foundation of NSFC in 2013, and the Award of the Program for New Century Excellent Talents funded by the Ministry of Education of the People's Republic of China in 2011.



Jun Wang (Member, IEEE) received the Ph.D. degree in electrical engineering from Columbia University, New York, NY, USA, in 2011. He is currently a Professor with the School of Computer Science and Software Engineering, East China Normal University, Shanghai, China and an Adjunct Faculty Member of Columbia University. From 2010 to 2014, he was a Research Staff Member with IBM T. J. Watson Research Center, Yorktown Heights, NY, USA. His research interests include machine learning, data mining, and computer vision. Dr. Wang has been the recipient of several awards, including the Outstanding Technical Achievement Award from IBM Corporation in 2013, and the Jury Thesis Award from Columbia University in 2011.